**4.DETAILED DESIGN**

**4.1 Introduction**

During detailed design, the internal logic of each modules specified in system design is decided. During this phase further details of the modules are decided. Design of each of the modules usually specified in a highlevel description language which is independent of the language in which software eventually be implemented.

**4.2 Structure of software package**



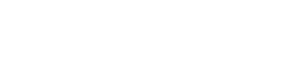
Network Algorithm Simulation



Parity Check

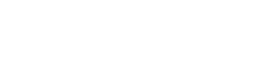


Checksum



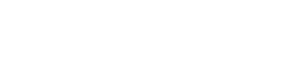
Cyclic Redundancy

Check



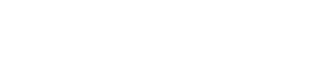
Simple Parity

Check



d Parity Check

2



Forward Error

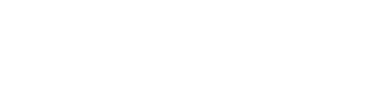
Correction



Hamming Code



Burst Error Correction



Shortest Path

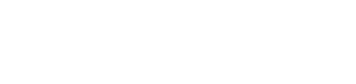
First(Dijkstra’s)



Open Shortest Path First



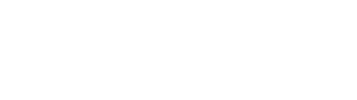
Border Gateway Protocol



Flooding

Routing

Algorithm



Distance Vector

Routing

Algorithm

**Figure 4.2 Structure of software package**

**4.3 Module decomposition of software**

**Structure Chart:**

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Name** | **Process** |
|  | **Data flow** | **Show the direction flow of data.** |
|  | **Control flow** | **Shows the direction of flow control.** |
|  | **Processing** | **Shows manipulation,**  **Calculation**  **and processing.** |
|  | **Module Invocation** | **It represent subordinate module being invoked by superior ordinate module.** |
| **C**  **B**  **A**  **Main** | **Condition invocation** | **It indicates that the invocation of subordinates**  **Module depends on the evaluation of a condition** |
| **B**  **A** | **Iteration** | **It represent the iteration** |

Structure chart is a top-down modular design, consist of squares representing different models in a system and lines. Structure chart shows how program has been partitioned into manageable modules hierarchy and organization of those modules and communicational interface.

**Table 4.1 Structure chart**

**Flow Chart:**

Flow chart is a graphical representation of solution to the given problems. A Flowchart is a pictorial representation of an algorithm, workflow or process. The diagrammatic representation illustrates a solution model to given problem. It uses the following symbol.

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Name** | **Purpose** |
|  | **Terminator** | **It indicates the start and end process.** |
|  | **Input/output** | **Input / output data** |
|  | **Decision** | **It represents a comparison or question that determines an alternative path to be followed.** |
|  | **Flow direction** | **Shows the direction of data flow.** |
|  | **Processing** | **It represents manipulation**  **Calculation or information**  **Processing.** |
|  | **Direction access storage** | **File storage** |
|  | **Preparation(Looping)** | **An instruction or Group of instruction** |

|  |  |  |
| --- | --- | --- |
|  | **In-Page** |  |
|  | **Off-Page** |  |
|  | **Delay** |  |

**Table 4.2 Flow Chart**

**4.3.1 Parity Check**

**4.3.1.1 Simple Parity Check**

**4.3.1.1.1 Input:**

**4.3.1.1.2 Procedural details:**

**Algorithm:**

Step 1:

Step 2:

Step 3:

Step 4:

Step 5:

Step 6:

**Algorithm 4.1.1.1 Simple Parity Check**

**4.3.1.1.3 File I/O Interface:**

**4.3.1.1.4 Outputs:**

**4.3.1.1.5 Implementation aspects:**

**4.3.1.2 2-D Parity Check**

**4.3.1.2.1 Input:**

**4.3.1.2.2 Procedural details:**

**Algorithm:**

Step 1:

Step 2:

Step 3:

Step 4:

Step 5:

Step 6:

**Algorithm 4.3.1.2 2-D Parity Check**